■ สุรศักดิ์ สงวนพงษ์

ไอพีรุ่นหก

ไอพีรุ่นหกยึดหลักพื้นฐานของไอพีเดิมไว้ โดยปรับปรุงบางส่วนให้มีประสิทธิภาพสูงขึ้น และสอดคล้องกับความต้องการทั้งในปัจจุบันและอนาคต

ไอพีรุ่นสี่กำลังประสบปัญหาแอดเดรสที่มีให้ไม่พอเพียงต่อการใช้งาน และตารางเลือกเส้นทางที่ขนาดใหญ่เกินกว่าที่เราเตอร์จะรองรับได้ ไออีทีเอฟวางแนวทางแก้ปัญหาโดยการพัฒนาโปรโตคอลใหม่ขึ้นมาทดแทนไอพีเพื่อเริ่มต้นไอพียุคใหม่หรือที่รู้จักในชื่อ ไอพีรุ่นหก

การใช้ไอพีแอดเดรสให้ได้ประสิทธิภาพร้อยเปอร์เซ็นต์เต็มตามจำนวนแอดเดรสทั้งหมดที่มีอยู่ประมาณสี่พันล้านแอดเดรสเป็นสิ่งที่ทำไม่ได้ในทางปฏิบัติ ปัญหาขาดแคลนไอพีแอดเดรสย่อมเกิดก่อนที่จะใช้แอดเดรสได้ทั้งหมด หรือถึงแม้จะมีวิธีใช้แอดเดรสได้อย่างเต็มประสิทธิภาพก็ตาม แอดเดรสไอพีรุ่นสี่ดูเหมือนจะไม่เพียงพอต่อการใช้งานในรูปแบบใหม่ ๆ เช่นคอมพิวเตอร์ในรถยนต์ หรือเซตท็อปบ็อกซ์สำหรับเว็บทีวี ความต้องการใช้แอดเดรสจำนวนมากเป็นสิ่งที่หลีกเลี่ยงไม่ได้ในอนาคต

ไออีทีเอฟเริ่มมองปัญหาขาดแคลนแอดเดรสและหาแนวทางแก้ไขมาตั้งแต่ปี 2524 ตลอดระยะเวลาที่ผ่านมามีผู้เสนอวิธีการหลากหลายรูปแบบ ไออีทีเอฟได้คัดเลือกข้อเสนอและกำหนดให้ไอพีในยุคถัดไปคือ ไอพีรุ่นหก (IP version 6 หรือ IPv6)

เนื่องจากแอดเดรสบรรจุอยู่ในเฮดเดอร์ของทุกแพ็กเกต การขยายขนาดแอดเดรสย่อมต้องเปลี่ยนแปลงขนาดเฮดเดอร์ และหมายถึงการปรับเปลี่ยนโปรโตคอลใหม่ ผลที่ตามมาอีกคือซอฟต์แวร์ในทุก ๆ โฮสต์และเราเตอร์ในอินเทอร์เน็ตต้องเปลี่ยนแปลงด้วยเช่นกัน การเปลี่ยนแปลงทั่วทั้งอินเทอร์เน็ตเป็นเรื่องใหญ่เกินกว่าที่จะทำได้ในระยะเวลาสั้น และยิ่งทำได้ยากหากไอพีรูปแบบใหม่มีความแตกต่างไปอย่างสิ้นเชิงจากรูปแบบเดิม

การเปลี่ยนแปลง

ในความเป็นจริงแล้วการปรับปรุงไอพีอาจใช้วิธีขยายขนาดแอดเดรสเพียงอย่างเดียว โดยไม่เปลี่ยนแปลงส่วนอื่นใดก็ได้ เพราะไอพีเดิมถือได้ว่าผ่านการออกแบบและปรับปรุงมาตามลำดับ แต่จากประสบการณ์การใช้ไอพีรุ่นสี่ตลอดระยะเวลาเกือบยี่สิบปีที่ผ่านมาช่วยชี้ให้เห็นถึงจุดด้อยบางจุดที่มีอยู่ในไอพีเดิม รวมทั้งความต้องการใช้งานแบบใหม่ที่จำเป็นในอนาคต ด้วยเหตุนี้ไอพีรุ่นหกจึงยังคงยึดหลักพื้นฐานของไอพีเดิมไว้โดยปรับปรุงบางส่วนให้มีประสิทธิภาพสูงขึ้นและสอดคล้องกับความ

ต้องการทั้งในปัจจุบันและอนาคต ไอพีรุ่นหกมีข้อปรับปรุงเพิ่มมาจากไอพีรุ่นสี่ดังต่อไปนี้

- ขยายขนาดแอดเดรสขึ้นเป็น 128 บิต
- เพิ่มขีดความสามารถการเลือกเส้นทาง และสนับสนุนโมบายโฮสต์ (mobile hosts)
- สนับสนุนการทำงานแบบเวลาจริง (real-time services)
- มีระบบติดตั้งแอดเดรสอัตโนมัติ (autoconfiguration)
- ปรับปรุงเฮดเดอร์เพื่อให้ประมวลผลได้มีประสิทธิภาพขึ้น
- เพิ่มระบบรักษาความปลอดภัย

ไอพีแอดเดรสรุ่นหก

ไอพีรุ่นหกมีแอดเดรสขนาด 128 บิต จำนวนแอดเดรสรวมประมาณ 3.4x1038 เทียบกับไอพีรุ่นสี่ที่มีแอดเดรสขนาด 32 บิต ซึ่งมีจำนวนแอดเดรสรวมเกือบสี่พันล้านแอดเดรส

ไอพีรุ่นหกเขียนแทนด้วยเลขฐานสิบหกจำนวนสี่หลักและใช้เครื่องหมาย ":" คั่นระหว่างตัวเลขดังตัวอย่างต่อไปนี้

```
FEDC:BA98:7654:3210:FEDC:BA98:7654:3210
```

เลขฐานสิบหกช่วยให้เขียนแอดเดรสได้กะทัดรัดและสะดวกกว่าเลขฐานสิบ แต่ระบบเลขฐานสิบหกและขนาดแอดเดรสที่เพิ่มขึ้นทำให้การจดจำยุ่งยากขึ้น ผู้ใช้ไอพีรุ่นหกจึงต้องพึ่งพาดีเอ็นเอสโดยไม่อาจหลีกเลี่ยงได้

เพื่อช่วยให้เขียนแอดเดรสได้สะดวกยิ่งขึ้น แอดเดรสบางแบบสามารถเขียนย่อได้ โดยเฉพาะแอดเดรสที่มีเลขศูนย์ต่อเนื่องกัน เช่น

```
1080:0000:0000:0000:0000:0008:0800:200C:417A
```

สามารถเขียนย่อโดยใช้ 0 ตัวเดียว แทน 0000 ได้ เช่น

```
1080:0:0:0:0:8:800:200C:417A
```

เลขศูนย์ที่ติดกันต่อเนื่องเป็นชุด สามารถใช้สัญลักษณ์ "::" แทนเลขศูนย์ทั้งชุดได้ เช่น จากตัวอย่างข้างต้นสามารถเขียนแอดเดรสได้ดังนี้

```
1080::0008:0800:200C:417A
```

เครื่องหมาย :: สามารถอยู่ได้ทั้งหน้าและหลังสุด รวมทั้งภายในเพื่อแทนเลขศูนย์ที่อยู่ติดกันได้ ดังตัวอย่างต่อไปนี้

```
FEDC:BA98::7654:3210          มีค่าเท่ากับ
FEDC:BA98:0:0:0:0:7654:3210
FEDC:BA98:7654:3210::         มีค่าเท่ากับ
FEDC:BA98:7654:3210:0:0:0:0
::FEDC:BA98:7654:3210         มีค่าเท่ากับ
0:0:0:0:FEDC:BA98:7654:3210
```

ในแอดเดรสหนึ่ง ๆ อนุญาตให้ใช้ "::" ได้เพียงครั้งเดียวเพื่อไม่ให้กำกวม ตัวอย่างเช่น 0:0:0:BA98:7654:0:0:0 อาจเขียนแทนด้วย ::BA98:7654:0:0:0 หรือ 0:0:0:BA98:7654:: อย่างใดอย่างหนึ่ง แต่ไม่สามารถเขียนแทนด้วย ::BA98:7654:: เนื่องจากไม่อาจแยกจำนวนบิตที่เป็นศูนย์ระหว่างส่วนหน้ากับส่วนหลังได้

แอดเดรสรุ่นสี่เขียนตามแบบรุ่นหก

แอดเดรสรุ่นสี่ที่ต้องเขียนในรูปของแอดเดรสรุ่นหกให้ใช้ 0 นำหน้า ตามด้วยรูปแบบเดิมของแอดเดรสรุ่นสี่ เช่น 10.0.0.1 ให้เขียนแทนด้วย ::10.0.0.1

การจัดสรรแอดเดรส

แอดเดรสรุ่นหกไม่จัดแบ่งออกเป็นคลาส แต่จะใช้พรีฟิกซ์ซึ่งมีขนาดไม่คงที่นำหน้าเพื่อจัดแบ่งแอดเดรสเป็นประเภทตามการใช้งาน ตารางที่ 1 แสดงการจัดสรรไอพีแอดเดรสรุ่นหก แอดเดรสส่วนใหญ่ยังเป็นแอดเดรสสำรอง พรีฟิกซ์ที่ได้จัดเตรียมไว้ เช่น

- พรีฟิกซ์ 0000 001 สำหรับโอเอสไอ
- พรีฟิกซ์ 0000 010 สำหรับโนเวลล์ไอพีเอ็กซ์
- พรีฟิกซ์ 001 เป็นพรีฟิกซ์สำหรับแอดเดรสในไอพีรุ่นหก แอดเดรสเดิมไม่ว่าจะเป็นคลาส เอ บี หรือ ซี จะถูกแทนที่ด้วยแอดเดรสพรีฟิกซ์นี้
- พรีฟิกซ์ลิงก์โลคอล (link local) และไซท์โลคอล (site local) เป็นพรีฟิกซ์ที่ใช้สำหรับแอดเดรสเฉพาะที่

แอดเดรสพิเศษเฉพาะกรณี

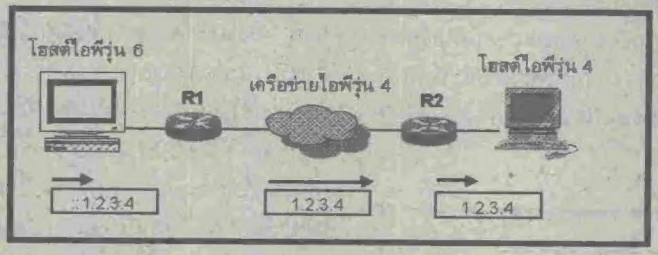
นอกไปจากแอดเดรสยูนิคาสต์ตามแบบที่ได้กล่าวมาแล้ว ไอพีรุ่นหกยังกำหนดยูนิคาสต์แอดเดรสพิเศษเป็นกรณีเฉพาะเพื่อใช้งานอีกดังนี้คือ

แอดเดรสไม่กำหนดแน่ชัด

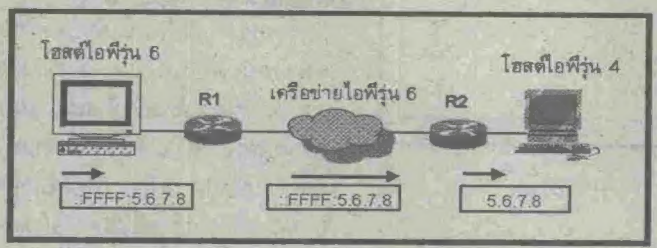
แอดเดรส 0:0:0:0:0:0:0:0 เรียกว่า แอดเดรสไม่กำหนดแน่ชัด (unspecified address) และเขียนโดยย่อได้คือ :: แอดเดรสนี้ไม่ใช้กำหนดให้กับเครื่องใดโดยตรง แต่ให้ใช้เป็นแอดเดรสต้นทางสำหรับสถานีที่ยังไม่มีแอดเดรสประจำตัวเพื่อสอบถามหาแอดเดรสที่แท้จริงต่อไป แอดเดรสนี้ไม่อนุญาตให้ใช้เป็นแอดเดรสปลายทางโดยเด็ดขาด

การจัดสรร	พรีฟิกซ์บิต	พรีฟิกซ์แอดเดรส	สัดส่วนการใช้
Reserved	0000 0000	0::/8	1/256
Unassigned	0000 0001	100::/8	1/256
Reserved for NSAP Allocation	0000 001	200::/7	1/128
Reserved for IPX Allocation	0000 010	400::/7	1/128
Unassigned	0000 011	600::/7	1/128
Unassigned	0000 1	800::/5	1/32
Unassigned	0001	1000::/4	1/16
Aggregatable Global Unicast Addresses	001	2000::/3	1/8
Unassigned	010	4000::/3	1/8
Unassigned	011	6000::/3	1/8
Unassigned	100	8000::/3	1/8
Unassigned	101	A000::/3	1/8
Unassigned	110	C000::/3	1/8
Unassigned	1110	E000::/4	1/16
Unassigned	1111 0	F000::/5	1/32
Unassigned	1111 10	F800::/6	1/64
Unassigned	1111 110	FC00::/7	1/128
Unassigned	1111 1110 0	FE00::/9	1/512
Link-Local Unicast Addresses	1111 1110 10	FE80::/10	1/1024
Site-Local Unicast Addresses	1111 1110 11	FEC0::/10	1/1024
Multicast Addresses	1111 1111	FF00::/8	1/256

ตารางที่ 1 การจัดสรรไอพีแอดเดรสรุ่นหก



รูปที่ 1 การใช้แอดเดรสไอพีรุ่นสี่ที่เข้ากับไอพีรุ่นหก



รูปที่ 2 การใช้แอดเดรสไอพีรุ่นสี่แบบแปลงสู่ไอพีรุ่นหก

แอดเดรสลูปแบ็ก

แอดเดรส 0:0:0:0:0:0:0:1 หรือเขียนในรูปย่อคือ ::1 ใช้เป็น แอดเดรสลูปแบ็ก (loopback address) เพื่อการส่งเดทาแกรมให้กับตนเอง

แอดเดรสรุ่นสี่ในแอดเดรสรุ่นหก

การปรับเปลี่ยนไปใช้ไอพีรุ่นหกไม่อาจกระทำในระยะเวลาอันสั้นทั่วทั้งอินเทอร์เน็ต ในทางปฏิบัติจึงต้องมีทั้งเครือข่ายที่ใช้ไอพีรุ่นสี่และรุ่นหกขนานกันไป ไอพีรุ่นหกจึงต้องจัดเตรียมรูปแบบแอดเดรสสำหรับทำงานร่วมกับไอพีรุ่นสี่ แอดเดรสที่จัดเตรียมไว้มีสองรูปแบบคือ แอดเดรสไอพีรุ่นหกที่เข้ากับไอพีรุ่นสี่ (IPv4-compatible IPv6 address) และ แอดเดรสไอพีรุ่นหกแปลงจากไอพีรุ่นสี่ (IPv4-mapped IPv6 address)

แอดเดรสไอพีรุ่นหกที่เข้ากับไอพีรุ่นสี่ออกแบบมาสำหรับการสื่อสารระหว่างสถานีที่ทำงานด้วยไอพีรุ่นหกอย่างเต็มตัว แต่การสื่อสารต้องผ่านเครือข่ายไอพีรุ่นสี่ เช่นในรูปที่ 1 แอดเดรสที่ใช้ในรูปแบบนี้จะใช้พรีฟิกซ์ค่าศูนย์จำนวน 96 บิต ต่อท้ายด้วยแอดเดรสไอพีรุ่นสี่ ตัวอย่างเช่น แอดเดรส 1.2.3.4 จะเปลี่ยนเป็น 0:0:0:0:0:0:1.2.3.4 หรือ ::1.2.3.4

รูปแบบที่สองคือแอดเดรสไอพีรุ่นหกแปลงจากไอพีรุ่นสี่ แอดเดรสนี้ออกแบบมาสำหรับสถานีไอพีรุ่นหกที่ต้องสื่อสารกับสถานีที่ทำงานด้วยไอพีรุ่นสี่อย่างเดียว รูปที่ 2 แสดงถึงภาวะการทำงานของแอดเดรสรูปแบบนี้ โดยเราเตอร์ปลายทางจะแปลงแพ็กเกตไอพีรุ่นสี่ไปสู่แพ็กเกตไอพีรุ่นหก แอดเดรสในรูปแบบนี้ใช้พรีฟิกซ์ค่าศูนย์จำนวน 80 บิต ตามด้วยพรีฟิกซ์ค่าหนึ่งจำนวน 16 บิต และต่อท้ายด้วยแอดเดรสไอพีรุ่นสี่ เช่นแอดเดรส 5.6.7.8 จะถูกเปลี่ยนเป็น 0:0:0:0:0:FFFF:5.6.7.8 หรือ ::FFFF:5.6.7.8

ไอพีเฮดเดอร์

เฮดเดอร์ไอพีรุ่นหกมีขนาดคงที่ 40 ไบต์ โดยประกอบด้วยฟิลด์กำหนดการทำงาน 8 ไบต์รวมกับแอดเดรสต้นทางและปลายทางอีก 32 ไบต์ตามรูปที่ 3 แต่ละฟิลด์มีความหมายดังต่อไปนี้

- Version ขนาด 4 บิต : แสดงรุ่นของโปรโตคอล มีค่าเท่ากับ 6
- Traffic class ขนาด 8 บิต : กำหนดกลุ่มแพ็กเกตตามคลาสและลำดับความสำคัญซึ่งยังอยู่ระหว่างการกำหนดรูปแบบและทดลองใช้
- Flow Label ขนาด 20 บิต : กำหนดรูปแบบการสื่อสาร
- Payload Length ขนาด 16 บิต : บอกขนาดข้อมูลในหน่วยไบต์โดยไม่รวมเฮดเดอร์ Next Header ขนาด 8 บิต : ทำหน้าที่เป็นทั้งตัวกำหนดชนิดโปรโตคอลและออปชันประจำเดทาแกรม หากใช้เป็นออปชันฟิลด์

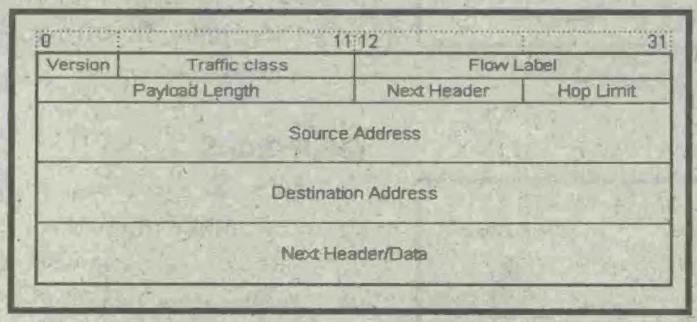
นี้จะบรรจุค่าแสดงเฮดเดอร์ของออปชันที่ใช้งานซึ่งบรรจุอยู่ต่อจากเฮดเดอร์ปกติ

- Hop Limit ขนาด 8 บิต : ใช้กำหนดระยะทางที่แพ็กเกตจะเดินทางผ่านได้ เราเตอร์ที่รับแพ็กเกตจะปรับลดค่าลงหนึ่งหน่วย หากลดลงเป็น 0 เราเตอร์จะกำจัดแพ็กเกตนั้น
- Source Address ขนาด 128 บิต : กำหนดไอพีแอดเดรสต้นทาง
- Destination Address ขนาด 128 บิต : กำหนดไอพีแอดเดรสปลายทาง

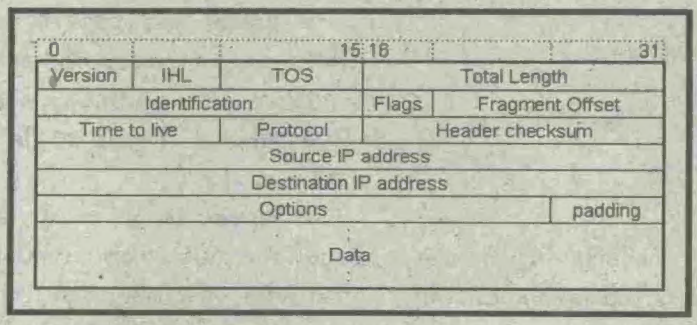
เปรียบเทียบเฮดเดอร์รุ่นสี่และรุ่นหก

ตามที่ได้กล่าวไว้แล้วในตอนต้นบทความว่าแรงจูงใจสำคัญในการสร้างไอพีรุ่นหกก็เพื่อรองรับการขยายตัวของอินเทอร์เน็ต ไอพีแอดเดรสเดิมที่มีขนาดเพียง 32 บิตได้ขยายขึ้นเป็น 128 บิต ขนาดของเฮดเดอร์ย่อมเปลี่ยนแปลงไปด้วย จึงเป็นจังหวะที่เหมาะสมต่อการปรับปรุงเฮดเดอร์ใหม่ไปพร้อมกัน

เมื่อพิจารณาเฮดเดอร์ของไอพีรุ่นหกตามรูปที่ 3 กับเฮดเดอร์ของไอพีรุ่นสี่ตามรูปที่ 4 จะพบว่าเฮดเดอร์ของไอพีรุ่นหกมีความเรียบง่ายกว่ามาก โดยลดจำนวนฟิลด์จาก 10 ฟิลด์ (ไม่นับแอดเดรสและออปชัน) เหลือเพียง 6 ฟิลด์



รูปที่ 3 เฮดเดอร์ไอพีรุ่นหก



รูปที่ 4 เฮดเดอร์ไอพีรุ่นสี่

เปรียบเทียบโดยสรุป

ฟิลด์ในเฮดเดอร์ของไอพีรุ่นหกที่คงตำแหน่งและความหมายเดิมมีเพียงฟิลด์เดียวคือฟิลด์ Version ขนาด 4 บิต เพื่อให้สามารถใช้งานทั้งไอพีรุ่นหกและรุ่นสี่ไปพร้อมกันได้โดยตรวจค่าในสี่บิตแรก โปรแกรมประยุกต์เพียงบรรจุค่า 4 หรือ 6 เพื่อกำหนดรุ่นของไอพี

ฟิลด์จำนวน 6 ฟิลด์ถูกกำจัดทิ้งไป ได้แก่ IHL, TOS, Identification, Flags, Fragment Offset และ Header Checksum ฟิลด์จำนวน 3 ฟิลด์เปลี่ยนชื่อและบางส่วนปรับการใช้งานไปเล็กน้อย ได้แก่ Total Length, Protocol และ Time To Live โดยเปลี่ยนไปเป็น Payload Length, Next Header และ Hop Limit ตามลำดับ

ฟิลด์ออปชันไม่ได้กำจัดทิ้งโดยตรงแต่ปรับรูปแบบการทำงานใหม่ทั้งหมดให้อยู่ภายใต้ฟิลด์ Next Header ฟิลด์ที่เพิ่มเข้ามาใหม่ 2 ฟิลด์ได้แก่ Traffic Class และ Flow Label

ขนาดเฮดเดอร์แบบคงที่

คุณสมบัติสำคัญของเฮดเดอร์ใหม่คือกำหนดขนาดคงที่ 40 ไบต์ ต่างจากเฮดเดอร์เดิมที่มีฟิลด์ออปชันซึ่งถือเป็นส่วนหนึ่งของเฮดเดอร์โดยตรงทำให้ขนาดเฮดเดอร์ไม่แน่นอน แต่เฮดเดอร์ของไอพีรุ่นหกยังคงอนุญาตให้มีออปชันได้โดยนำค่าออปชันมาต่อท้ายเฮดเดอร์หลักและใช้ฟิลด์ Next Header บ่งบอกว่ามีออปชันหรือไม่ วิธีนี้ช่วยแยกออปชันให้เป็นอิสระจากเฮดเดอร์ และไม่จำเป็นต้องใช้ฟิลด์ IHL เพื่อบอกขนาดเฮดเดอร์อีกต่อไป

กำจัดค่าผลรวมตรวจสอบ

การกำจัดฟิลด์ผลรวมตรวจสอบช่วยให้ไอพีมีสมรรถนะดีขึ้นเนื่องจากลดขั้นตอนการคำนวณค่าผลรวมในแต่ละแพ็กเกต ถึงแม้ว่าในแง่มุมหนึ่งอาจเสี่ยงต่อความผิดพลาดที่เกิดขึ้นเช่นส่งแพ็กเกตผิดเส้นทาง แต่กลุ่มผู้ออกแบบไอพีรุ่นหกก็เลือกที่จะกำจัดฟิลด์นี้ออกไป เพราะเครือข่ายระดับเดทาลิงก์จะต้องเอ็นแคปซูเลตและคำนวณค่าผลรวมตรวจสอบอยู่แล้ว

ห้ามแฟรกเมนต์ที่เราเตอร์ระหว่างทาง

ไอพีรุ่นสี่อนุญาตให้แฟรกเมนต์แพ็กเกตได้ทั้งในโฮสต์ที่ส่งแพ็กเกตและที่เราเตอร์ระหว่างทาง แพ็กเกตที่ผ่านการแฟรกเมนต์แล้วยังอาจถูกแฟรกเมนต์ต่อไปได้อีก แต่ในไอพีรุ่นหกอนุญาตให้เกิดแฟรกเมนต์ได้เฉพาะที่โฮสต์ต้นทางซึ่งส่งแพ็กเกตเท่านั้น โดยใช้เฮดเดอร์ส่วนขยาย เราเตอร์ระหว่างทางไม่มีสิทธิ์แฟรกเมนต์แพ็กเกต

ทั้งไอพีรุ่นสี่และไอพีรุ่นหกมีกลไกตรวจหา เอ็มทียูตลอดเส้นทาง (Path MTU Discovery) เพื่อหาเอ็มทียูที่เล็กที่สุดประจำเส้นทางและใช้เอ็มทียูนั้นจัดขนาดแพ็กเกตให้เหมาะสมโดยไม่ต้องแฟรกเมนต์ วิธีนี้ไม่ได้บังคับใช้ในไอพีรุ่นสี่แต่เป็นข้อบังคับในไอพีรุ่นหก ด้วยหลักการนี้ฟิลด์ Identification, Flags และ Fragment Offset ที่ใช้ในการแฟรกเมนต์จึงไม่ปรากฏอยู่ในเฮดเดอร์

ปรับรูปแบบฟิลด์กำหนดขนาด

ไอพีรุ่นสี่ใช้ฟิลด์ Total Length กำหนดขนาดรวมของเฮดเดอร์และข้อมูล แต่สำหรับไอพีรุ่นหกใช้ฟิลด์ Payload Length บอกเฉพาะขนาดข้อมูลเท่านั้น เนื่องจากเฮดเดอร์มีขนาดคงที่ 40 ไบต์

ยกเลิกการกำหนดประเภทการให้บริการ

ฟิลด์ TOS กำหนดประเภทการให้บริการว่าควรส่งแพ็กเกตไปในเส้นทางที่เร็วกว่า สั้นกว่า หรือน่าเชื่อถือกว่า แต่ในทางปฏิบัติแล้วโปรแกรมประยุกต์ส่วนใหญ่ไม่ได้ใช้ฟิลด์นี้แต่อย่างใด ไอพีรุ่นหกจึงกำจัดฟิลด์นี้ออกไป และวางแนวทางการใช้บริการแบบใหม่โดยใช้ฟิลด์ Traffic class และ Flow Label

เปลี่ยนชื่อฟิลด์ค่าจำกัดระยะทาง

ไอพีรุ่นสี่ใช้ฟิลด์นี้กำหนดอายุของแพ็กเกตในหน่วยวินาที โดยลดค่าลง 1 วินาที หากแพ็กเกตอยู่ในคิวเกิน 1 วินาที แต่ในทางปฏิบัติเป็นเรื่องยากที่จะวัดเวลาที่แต่ละแพ็กเกตรอคอยอยู่ในคิว เราเตอร์ส่วนใหญ่จึงใช้วิธีลดค่า TTL ครั้งละหนึ่งทุกครั้งที่แพ็กเกตเดินทางผ่าน ไอพีรุ่นหกจึงเปลี่ยนชื่อฟิลด์ให้สื่อความหมายตรงกับการใช้งาน

ปรับรูปแบบฟิลด์กำหนดโปรโตคอล

ฟิลด์ Protocol จากไอพีรุ่นสี่เปลี่ยนชื่อเป็น Next Header พีดียูเช่นทีพีซีเซกเมนต์หรือยูดีพีเดทาแกรมในไอพีรุ่นสี่จะอยู่ต่อจากเฮดเดอร์ทันที แต่ไอพีรุ่นหกในกรณีที่มีออปชัน ค่าออปชันจะแทรกอยู่ระหว่างเฮดเดอร์กับพีดียู แต่ในกรณีที่ไม่มีออปชัน ค่าพีดียูจะอยู่ต่อจากเฮดเดอร์ทันที ฟิลด์ Next Header จะทำงานรองรับทั้งสองกรณีนี้รวมกันโดยกำหนดค่าเพื่อบอกว่าถัดจากเฮดเดอร์ไปแล้วเป็นออปชันหรือพีดียู

ฟิลด์ใหม่

ฟิลด์ใหม่ในไอพีรุ่นหกมี 2 ฟิลด์คือ Traffic class และ Flow Label ฟิลด์ Traffic class ออกแบบมาเพื่อสนับสนุนการทำงานแบบเวลาจริง ส่วน Flow Label ใช้กำหนดวิธีดำเนินการเฉพาะให้กับสายแพ็กเกตที่ต้องการ

จุดเด่นอื่นในไอพีรุ่นหก

นอกจากการขยายแอดเดรสแล้ว ไอพีรุ่นหกมีจุดเด่นอื่นที่รองรับการทำงานในอนาคตไม่ว่าจะเป็นเรื่องการสนับสนุนการทำงานแบบเวลาจริงโดยใช้โฟลว์เลเบลหรือการขยายขนาดออปชัน จุดเด่นสำคัญอีกสองเรื่องในไอพีรุ่นหก ได้แก่ การติดตั้งอัตโนมัติ และระบบรักษาความปลอดภัย

การติดตั้งแบบอัตโนมัติ

ในการติดตั้งแอดเดรสในไอพีรุ่นสี่ตามปกติ ผู้ใช้ป้อนแอดเดรสและพารามิเตอร์อื่น ๆ ที่จำเป็น ถึงแม้ว่าจะใช้ดีเอชซีพีช่วยในการติดตั้งระบบแต่ก็ไม่จัดอยู่ในข่ายการติดตั้งแบบอัตโนมัติที่สมบูรณ์แบบ รูปแบบใหม่ของการติดตั้งอัตโนมัติในไอพีรุ่นหกคือการเชื่อมเครื่องเข้าสู่เครือข่ายแบบ "Plug and Play" เครื่องสามารถกำหนดแอดเดรสพร้อมทั้งพารามิเตอร์อื่นที่จำเป็นโดยผู้ใช้ไม่ต้องดำเนินการใด

รูปแบบที่จัดเตรียมไว้สำหรับระบบติดตั้งอัตโนมัติในไอพีรุ่นหกจัดได้เป็น 3 แบบคือ แบบเฉพาะที่ (local scope model) แบบไร้สถานะ (stateless model) และแบบเต็มสถานะ (stateful server model)

การติดตั้งอัตโนมัติแบบเฉพาะที่

การติดตั้งอัตโนมัติแบบเฉพาะที่คือการใช้ลิงก์โลคอลแอดเดรสด้วยพรีฟิกซ์ FE80::/10 ต่อท้ายด้วยแมคแอดเดรส รูปแบบนี้ออกแบบมาใช้งานกับเครือข่ายที่ไม่มีเราเตอร์ หรือเครือข่ายขนาดเล็กที่ไม่ได้ต่อเชื่อมกับเครือข่ายอื่นใด

การติดตั้งอัตโนมัติแบบไร้สถานะ

การติดตั้งอัตโนมัติแบบไร้สถานะใช้หลักการของมัลติคาสต์ โดยไม่ต้องปรับระบบที่โฮสต์ที่ขอติดตั้งแต่อย่างใด รวมทั้งไม่จำเป็นต้องมีเซิร์ฟเวอร์ให้บริการเพื่อกำหนดค่าการติดตั้ง หากแต่ต้องเตรียมเราเตอร์ที่สนับสนุนระบบนี้ไว้

สถานีที่ต่อเข้าเครือข่ายจะส่งแพ็กเกตร้องขอโดยใช้เป็นลิงก์โลคอลแอดเดรสเป็นแอดเดรสต้นทาง และใช้มัลติคาสต์แอดเดรส FF02::2 เป็นแอดเดรสปลายทาง พร้อมทั้งส่งพารามิเตอร์ประจำตัวเครื่องเช่นแมคแอดเดรสหรือค่าอื่น ๆ ไปด้วย

Networking

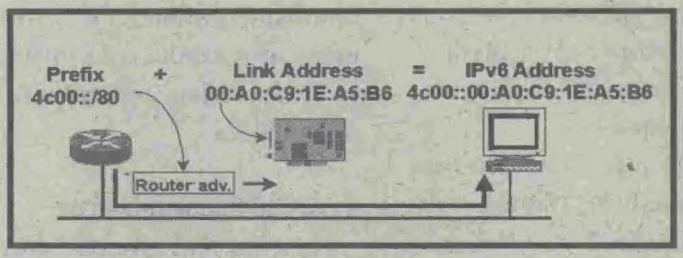
เราเตอร์ที่ให้บริการแอดเดรสจะรับแพ็กเกตร้องขอนั้นและประกาศพรีฟิกซ์บิตที่กำหนดขับเน็ตออกไป แอดเดรสที่ได้ประจำเครื่องจะเกิดจากการรวมระหว่างพรีฟิกซ์และแอดเดรสอินเตอร์เฟสของสถานีดังรูปที่ 5 การทำงานแบบนี้เรียกว่าไร้สถานะเพราะเราเตอร์ไม่มีข้อมูลของเครื่องใดล่วงหน้า ระบบนี้จึงเรียบง่ายและสะดวกต่อการดูแล

การติดตั้งอัตโนมัติแบบเต็มสถานะ

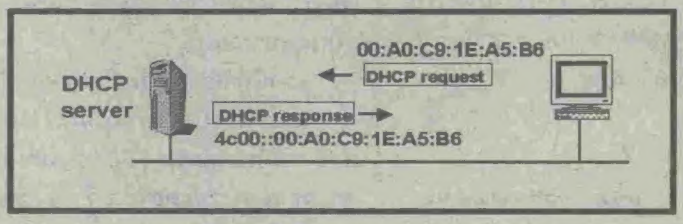
การติดตั้งอัตโนมัติแบบเต็มสถานะอาศัยเซิร์ฟเวอร์ซึ่งเก็บข้อมูลของเครื่องที่จะให้บริการไว้ก่อนหน้า ฐานข้อมูลในเซิร์ฟเวอร์จะกำหนดแอดเดรสให้กับโฮสต์เมื่อมีแพ็กเกตร้องขอส่งมา เซิร์ฟเวอร์จะตรวจหาจากฐานข้อมูลและส่งแอดเดรสกลับไป กระบวนการและรูปแบบการติดตั้งจะอาศัยโปรโตคอลดีเอชซีพีดังรูปที่ 6

เปรียบเทียบรูปแบบ

การติดตั้งแบบไร้สถานะใช้เมื่อไม่ต้องกำหนดแอดเดรสให้สถานีอย่างเฉพาะเจาะจง ทั้งนี้เพราะในกระบวนการนี้สถานีหนึ่ง ๆ อาจได้แอดเดรสแต่ละครั้งของการร้องขอไม่เหมือนเดิม ในขณะที่แบบเต็มสถานะใช้เมื่อต้องการควบคุมให้สถานีหนึ่ง ๆ มีแอดเดรสเฉพาะเจาะจง ในเครือข่ายหนึ่ง ๆ สามารถใช้การติดตั้งทั้งสองแบบพร้อมกันได้



รูปที่ 5 การติดตั้งอัตโนมัติแบบไร้สถานะ



รูปที่ 6 การติดตั้งอัตโนมัติแบบเต็มสถานะ

การรักษาความปลอดภัย

ไอพีรุ่นหกมีระบบรักษาความปลอดภัยสองรูปแบบหลักคือ การพิสูจน์ตัวจริง (authentication) และ การเข้ารหัสลับ (encryption)

การพิสูจน์ตัวจริงเป็นกระบวนการพิสูจน์ว่าแพ็กเกตมาจากผู้ส่งต้นทางตัวจริง ระบบจึงช่วยยืนยันว่าไม่มีการปลอมหรือแอบอ้างผู้ส่ง ส่วนการเข้ารหัสลับเป็นการเข้ารหัสข้อมูล เพื่อให้เฉพาะผู้รับปลายทางเท่านั้นที่สามารถถอดรหัสและอ่านข้อมูลได้

การพิสูจน์ตัวจริงในไอพีรุ่นหกใช้วิธีสร้างค่าพิสูจน์ตัวจริงโดยคำนวณจากข้อมูลที่ต้องการส่ง ผู้รับต้นปลายทางจะนำข้อมูลมาคำนวณค่าพิสูจน์ตัวจริงและเปรียบเทียบกับค่าที่ได้รับ หากมีค่าตรงกันก็ถือว่าแพ็กเกตนั้นส่งมาจากต้นทางจริง

ไอพีรุ่นหกกำหนดให้ใช้ขั้นตอนวิธีแบบ เอ็มดีไฟว์ (MD5 : Message Digest 5) เป็นขั้นตอนวิธีโดยปริยายสำหรับใช้คำนวณค่าพิสูจน์ตัวจริง และใช้เฮดเดอร์ชนิด 51 เป็นออปชันสำหรับส่งค่าพิสูจน์ตัวจริงไปพร้อมกับข้อมูล ไอพีรุ่นหกกำหนดให้โฮสต์ต้องสนับสนุนการพิสูจน์ตัวจริงและใช้เอ็มดีไฟว์เป็นขั้นตอนวิธีโดยปริยาย แต่โฮสต์สองฝั่งสามารถตกลงเพื่อเลือกใช้ขั้นตอนวิธีอื่นได้เช่นกัน การพิสูจน์ตัวจริงไม่ใช่ข้อบังคับใช้ หากโฮสต์เห็นว่าไม่ต้องพิสูจน์ก็ไม่จำเป็นต้องใช้แต่อย่างใด

การเข้ารหัสลับ

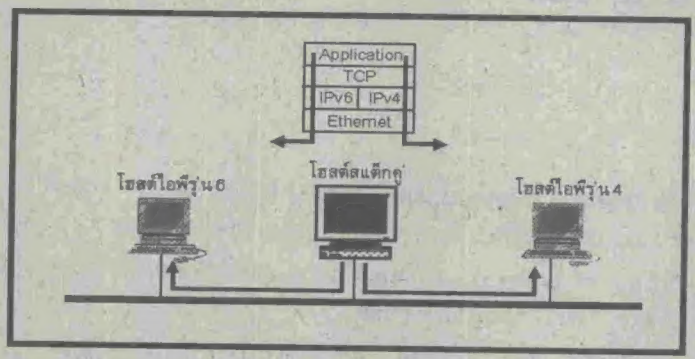
การพิสูจน์ตัวจริงเป็นเพียงการรับประกันว่าข้อมูลมีความถูกต้องและส่งมาจากผู้ส่งตัวจริง แต่ไม่อาจป้องกันการดักจับแพ็กเกตและการลักลอบนำข้อมูลไปใช้ระหว่างที่แพ็กเกตต้องเดินทางผ่านเครือข่ายหลายเครือข่าย วิธีป้องกันการลักลอบนำข้อมูลไปใช้จำเป็นต้องใช้การเข้ารหัสลับ ข้อมูลจะผ่านขั้นตอนวิธีเข้ารหัสทำให้ถึงแม้เปิดดูข้อมูลได้แต่ก็ไม่สามารถถอดรหัสเพื่อดูข้อมูลจริงได้หากไม่มีกุญแจรหัส ไอพีรุ่นหกกำหนดให้ใช้วิธีเข้ารหัสด้วย ซีบีซี (CBC : Cipher Block Chaining) ตามแบบที่กำหนดด้วยขั้นตอนวิธีดีอีเอส (DES : Data Encryption Standard)

การปรับเปลี่ยนไปสู่ไอพีรุ่นหก

การปรับเปลี่ยนจากไอพีรุ่นสี่ไปเป็นรุ่นหกจะเกิดขึ้นเมื่อใดยังไม่มีใครให้คำตอบนี้ได้ แต่ไออีทีเอฟคาดการณ์ว่าแอดเดรสรุ่นสี่จะรองรับการใช้งานได้ไปจนกระทั่งราว ค.ศ. 2010 การเปลี่ยนโฮสต์หรือเราเตอร์ไปสู่ไอพีรุ่นหกพร้อมกันทั่วโลกเป็นสิ่งที่เป็นไปไม่ได้ ในทางตรงกันข้าม

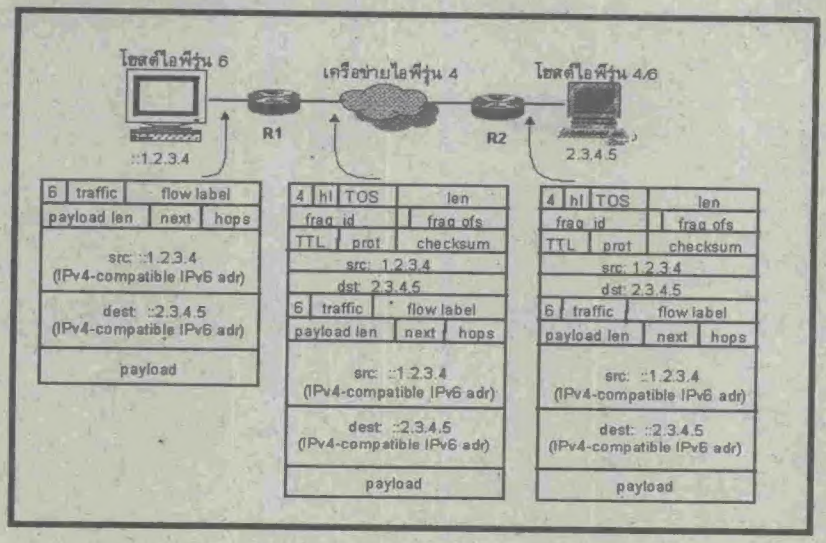
การใช้ไอพีรุ่นหกจะต้องเป็นการปรับเปลี่ยนไปทีละส่วน เครือข่ายที่มีความพร้อมสามารถใช้ไอพีรุ่นหกทั้งเครือข่ายได้ บางเครือข่ายอาจมีโฮสต์และเราเตอร์ที่ใช้ทั้งไอพีรุ่นสี่และรุ่นหกพร้อมกัน เครือข่ายที่ยังไม่พร้อมก็ยังสามารถใช้ไอพีรุ่นสี่ต่อไปได้

ไออีทีเอฟกำหนดมาตรการรองรับการปรับเปลี่ยน เช่น ให้โฮสต์และเราเตอร์ทำงานทั้งไอพีรุ่นสี่และรุ่นหกได้พร้อมกันโดยใช้ สแต็กคู่ (dual stack) หรือใช้ อุโมงค์เครือข่าย (tunneling) โดยเอ็นแคปซูเลตแพ็กเกตไอพีรุ่นหกในแพ็กเกตไอพีรุ่นสี่ ในปัจจุบันมีโครงการ "6bone" ต่อเชื่อมระหว่างเครือข่ายไอพีรุ่นหกโดยใช้อุโมงค์



รูปที่ 7
โหนดแบบสแต็กคู่สามารถทำงานได้ทั้งไอพีรุ่นหกและไอพีรุ่นสี่

รูปที่ 8
การเอ็นแคปซูเลตในอุโมงค์อัตโนมัติ



การใช้สแต็กคู่

วิธีการโดยตรงที่ให้โหนดไอพีรุ่นหกยังคงทำงานร่วมกับโหนดไอพีรุ่นสี่ได้คือให้โหนดไอพีรุ่นหกมีฟังก์ชันการทำงานตามไอพีรุ่นสี่อยู่ภายในตัว หากมองในรูปแบบการทำงานเป็นเลเยอร์แล้วแต่ละโหนดจะมีโปรโตคอลสแต็กรองรับทั้งรุ่นหกและรุ่นสี่ หรือมีสแต็กคู่อยู่ในตัว ตามรูปที่ 7 เมื่อโหนดที่ทำงานด้วยสแต็กคู่ต้องการติดต่อกับโหนดรุ่นหกก็เลือกใช้สแต็กด้านไอพีรุ่นหก และใช้สแต็กด้านไอพีรุ่นสี่เมื่อต้องติดต่อกับโหนดรุ่นสี่

ด้วยวิธีนี้โหนดจะต้องมีแอดเดรสทั้งรุ่นหกและรุ่นสี่ โดยสามารถใช้แอดเดรสรุ่นสี่ตามเดิมได้ และเพิ่มแอดเดรสรุ่นหกที่เข้ากันได้กับรุ่นสี่คือใช้พรีฟิกซ์ศูนย์จำนวน 96 บิต ต่อท้ายด้วยแอดเดรสรุ่นสี่ ตัวอย่างเช่น แอดเดรส 1.2.3.4 จะได้แอดเดรสรุ่นหกเท่ากับ 0:0:0:0:0:0:1.2.3.4 หรือ ::1.2.3.4

การทำงานแบบสแต็กคู่ต้องอาศัยดีเอ็นเอสเพื่อแปลงชื่อเป็นแอดเดรส ดีเอ็นเอสจะใช้เรคอร์ด "AAAA" เพื่อระบุว่าเครื่องที่ใช้งานมีแอดเดรสรุ่นหก ส่วนโหนดที่ทำงานด้วยไอพีรุ่นสี่ยังคงใช้เรคอร์ด A ตามปกติ เช่น

node-v6	AAAA	0:0:0:0:0:0:1.2.3.4
node-v4	A	1.2.3.4

การใช้อุโมงค์เครือข่าย

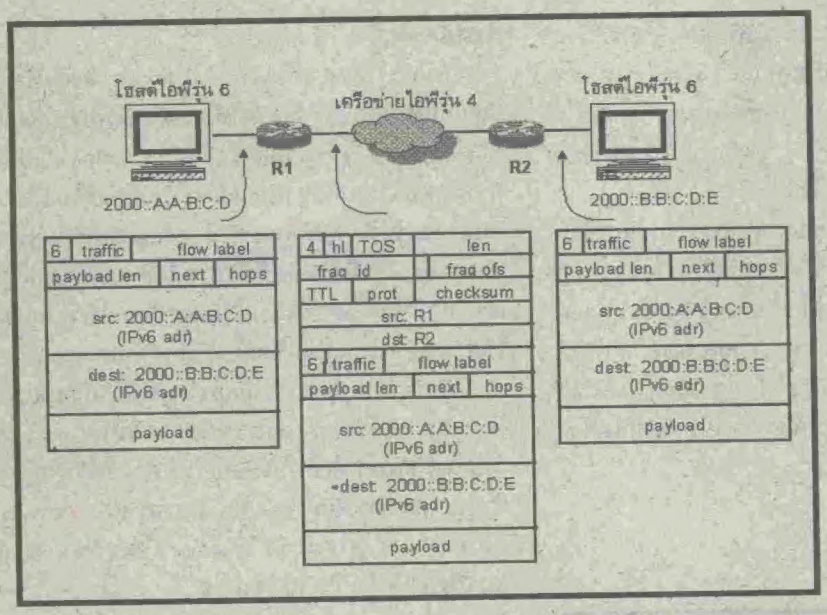
ถึงแม้ว่าการใช้สแต็กคู่จะช่วยให้สามารถใช้ไอพีรุ่นหกได้ แต่เมื่อต้องส่งแพ็กเกตข้ามเครือข่าย เครือข่ายที่เป็นทางผ่านของแพ็กเกตต้องสนับสนุนการทำงานของไอพีรุ่นหกหรือทำงานแบบสแต็กคู่ได้เช่นกัน แต่ในช่วงการปรับเปลี่ยนอาจมีเครือข่ายที่ยังไม่สนับสนุนการทำงานไอพีรุ่นหก การส่งแพ็กเกตไอพีรุ่นหกข้ามเครือข่ายไอพีรุ่นสี่จึงต้องอาศัยอุโมงค์

เพื่อให้เห็นขั้นตอนของการเอ็นแคปซูเลตแพ็กเกตตามวิธีการใช้อุโมงค์ทั้งสองแบบจะยกตัวอย่างการเชื่อมโยงระหว่างโฮสต์ผ่านเราเตอร์ตามหัวข้อย่อยต่อไปนี้

ตัวอย่างอุโมงค์อัตโนมัติ

รูปที่ 8 แสดงการเอ็นแคปซูเลตแพ็กเกตแบบอุโมงค์อัตโนมัติ แพ็กเกตไอพีรุ่นหกที่เดินทางมาถึงขอบเครือข่ายและเตรียมเข้าสู่เครือข่ายไอพีรุ่นสี่ แพ็กเกตจะถูกเอ็นแคปซูเลตเข้าสู่รูปแบบของไอพีรุ่นสี่ เราเตอร์ต้องแยกเอาแอดเดรสรุ่นสี่ออกจากแอดเดรสรุ่นหกเพื่อนำมาใช้ในเฮดเดอร์ไอพีรุ่นสี่ กระบวนการทั้งหมดไม่จำเป็นต้องติดตั้งระบบใดเพิ่มเติม

Networking



รูปที่ 9
การเอ็นแคปซูเลตในอุโมงค์แบบติดตั้ง

ตัวอย่างอุโมงค์แบบติดตั้ง

หากแอดเดรสรุ่นหกไม่ได้อยู่ในรูปแบบแอดเดรสรุ่นหกที่เข้ากันได้กับแอดเดรสรุ่นสี่ เราเตอร์จะไม่สามารถแยกแอดเดรสรุ่นสี่ออกมาได้ วิธีแก้ปัญหานี้ต้องจัดเตรียมเราเตอร์ดังรูปที่ 9 โดยให้ต้นทาง R1 ทราบว่าจะต้องสร้างอุโมงค์ไปยังเราเตอร์ปลายทางที่ R2 (เอ็นแคปซูเลตไอพีรุ่นหกในไอพีรุ่นสี่ไปยัง R2) และ R2 ถอดแพ็กเกตไอพีรุ่นหกเพื่อส่งต่อไปยังโฮสต์ปลายทาง

เว็บไซต์แนะนำ

- http://www.ipv6.org : โฮมเพจการใช้และพัฒนาไอพีรุ่นหก
- http://www.6bone.net : โฮมเพจการเชื่อมเครือข่ายไอพีรุ่นหกด้วยการใช้อุโมงค์
- http://www.ietf.org/html.charters/ipngwg-charter.html : กลุ่มงานพัฒนาไอพีรุ่นหกของไออีทีเอฟ
- http://www.ipv6forum.com : โฮมเพจข้อมูลไอพีรุ่นหก
- http://www.ipv6.org/mailing-lists.html : เมลลิงลิสต์สำหรับผู้สนใจไอพีรุ่นหก
- http://www.playground.sun.com/pub/ipng/html/ipng-main.html : ข้อมูลไอพีรุ่นหก
- http://www.3com.com/nsc/ipv6.html : ข้อมูลไอพีรุ่นหก
- http://www.bieringer.de/linux/IPv6/IPv6-HOWTO/IPv6-HOWTO.html : ไอพีรุ่นหกสำหรับลีนุกซ์
- http://www.ietf.org/html.charters/ipsec-charter.html : โฮมเพจกลุ่มงานกำหนดมาตรฐานโปรโตคอลความปลอดภัยสำหรับไอพี